

Social Analytics Web Interface v2.0

Resumen de la aplicación:

Social Analytics Web Interface v2.0 es una aplicación web que permite visualizar y gestionar gráficas estadísticas avanzadas sobre contenido de las redes sociales, tanto a nivel histórico como en tiempo real. Esto es posible gracias al consumo de servicios webs estándares que ofrecen metadatos en formato JSON sobre analítica social, como es el caso de *Social Analytics Process v2.0* (ver publicación en RUA).

Social Analytics Web Interface v2.0 ofrece a nivel general las siguientes funcionalidades:

- Visualización de analítica social a nivel genérico donde se muestran gráficas estadísticas
- Visualización de comparativa entre entidades analizadas en relación con el contenido social que ha sido procesado
- Analítica tanto para histórico de datos como para datos en tiempo real en las dos funcionalidades previamente mencionadas
- Analítica de datos en línea temporal de cada aspecto recuperado durante el procesamiento de datos.
- Gestión de correos de notificaciones donde se envían analíticas configuradas en formato pdf para períodos de tiempo específicos
- Gestión de usuarios basados en identificación abierta con Twitter e Instagram
- Gestión de analíticas y entidades
- Gestión de informes generados en formato pdf
- Gestión de datos procesados con tecnologías del lenguaje humano en formato CSV para su posterior análisis
- Consulta de metadatos en formato JSON
- Gestión de monitorizaciones de contenido social en diferentes idiomas. En particular inglés, español, francés y catalán.
- Servicios Web RESTful que gestionan los datos almacenados en el *Social Analytics Database v2.0* (ver publicación en RUA) y sirven de puente para gestionar la información presentada en la interfaz visual

Esta interfaz web, entre otras cosas, ofrece un acceso adaptable a los diferentes dispositivos que acceden a ella, permitiendo de esta forma mostrar en todo momento una visualización apropiada para su lectura, comprensión e interacción con las diferentes funcionalidades permitidas.

Social Analytics Web Interface v2.0 a diferencia de versiones anteriores, incluye correcciones de procesamiento y presentación de contenido sumado a nuevas funcionalidades añadidas que completan

el listado anterior. Además, se añaden funcionalidades de filtrado de listas (i.e. usuarios, colecciones, licencias) para simplificar los procesos de gestión.

Lenguaje de programación: Java

Entorno Operativo: multiplataforma (Windows, Linux, Mac OS X, etc.)

Listado de librerías fuente:

- Directorio src/main/java/es/gplsi/socialanalytics/exception:
 - AuthenticationRequiredException.java
 - ErrorMessage.java
 - LoginException.java
 - LoginExceptionMapper.java
 - NotAllowedException.java
 - PasswordIncorrectException.java
 - UserDoesNotExistException.java
- Directorio src/main/java/es/gplsi/socialanalytics/filter:
 - UrlRewriteFilter.java
- Directorio src/main/java/es/gplsi/socialanalytics/model:
 - DateHistogram.java
 - DateRange.java
 - Filters.java
 - TermCount.java
 - UserCount.java
 - TermCountHistogram.java
 - UserCount.java
 - UserCountHistogram.java
- Directorio src/main/java/es/gplsi/socialanalytics/report:
 - MailReportGenerator.java
 - PDFReportCreator.java
 - URLParameterEncoder.java
 - WebScraper.java
- Directorio src/main/java/es/gplsi/socialanalytics/resource:
 - AuthenticationResource.java
 - AuthenticationServiceProvider.java
 - CollectionResource.java
 - CollectionSearchResource.java

- CollectionServiceProvider.java
- CSVReportServiceProvider.java
- EntityResource.java
- NotificationServiceProvider.java
- PromotedCollectionResource.java
- StatisticsResource.java
- StatisticsServiceProvider.java
- CSVReportServiceProvider.java
- Directorio src/main/java/es/gplsi/socialanalytics/service:
 - AuthenticationService.java
 - AuthenticationServiceMysql.java
 - CollectionService.java
 - CollectionServiceMysql.java
 - NotificationService.java
 - NotificationServiceMysql.java
 - StatisticsService.java
 - StatisticsServiceElasticsearch.java
- Directorio src/main/java/es/gplsi/socialanalytics/servlet:
 - AnalyticsServlet.java
 - AuthenticationServlet.java
 - HomeServlet.java
 - IndexServlet.java
 - StorageServlet.java
- Directorio src/main/java/es/gplsi/socialanalytics:
 - SocialAnalyticsApplication.java
 - SocialAnalyticsListener.java
- Directorio src/main/java/es/gplsi/socialanalytics/util:
 - AuthenticationUtils.java
 - ParametersUtils.java
- Directorio src/main/webapp/app:
 - app.module.js
 - app.routes.js
- Directorio src/main/webapp/app/dashboard:
 - account.delete.html
 - collection.add.html
 - collection.delete.html
 - collection.edit.html

- collection.filters.html
 - csvReport.delete.html
 - currentAccount.delete.html
 - dashboard.html
 - dashboard.js
 - dashboard.module.js
 - entity.add.html
 - entity.delete.html
 - entity.edit.html
 - notification.delete.html
 - notification.edit.html
 - user.delete.html
 - user.edit.html
 - user.filters.html
- Directorio src/main/webapp/app/home:
 - home.html
 - home.js
 - home.module.js
- Directorio src/main/webapp/app/login:
 - login.html
 - login.js
 - login.module.js
- Directorio src/main/webapp/app/policy:
 - privacypolicy.html
 - privacypolicy.js
 - privacypolicy.module.js
- Directorio src/main/webapp/app/redirect:
 - redirect.js
 - redirect.module.js
- Directorio src/main/webapp/app/services:
 - authentication.service.js
 - collection.service.js
 - data.service.js
 - loading.service.js
 - services.module.js
 - stats.service.js

- utils.service.js
- Directorio src/main/webapp/app/signup:
 - signup.html
 - signup.js
 - signup.module.js
- Directorio src/main/webapp/app/stats:
 - notification.add.html
 - csvReport.add.html
 - notificationReport.edit.html
 - params.js
 - stats.html
 - stats.js
 - stats.module.js
 - versus.html
 - versus.js
- Directorio src/main/webapp/app/stats/views
 - compare.html
 - compare.js
 - emotion.js
 - histogram.js
 - histogram-terms.js
 - number.html
 - number.js
 - polarity.js
 - ranking.html
 - ranking.js
 - source.js
 - terms.html
 - Terms.js
 - urls.js
 - urls.html
 - users.js
- Directorio src/main/webapp/assets/css:
 - analytics.scss
 - common.scss

- dashboard.scss
 - home.scss
 - login.scss
 - stats.scss
- Directorio src/main/webapp/assets/img:
 - socialanalytics-default.png
 - socialanalytics01.png
 - socialanalytics02.png
 - user.png
- Directorio src/main/webapp/WEB-INF:
 - config.properties
- Directorio src/main/webapp/WEB-INF/jsp:
 - analytics.jsp
 - home.jsp
 - index.jsp
 - redirect.jsp
 - urlrewrite.xml
 - web.xml

Requerimientos:

Oracle Java 8

- Tomcat 7
- MariaDB 5.1 (o superior)
- Elasticsearch 2
- Social Analytics Listener
- Social Analytics Process

Instalación:

Antes comenzar a utilizar la aplicación, es necesario asegurarse de que los servidores Tomcat, MariaDB y Elasticsearch están funcionando. Además, para poder añadir y visualizar datos, es necesario instalar y ejecutar Social Analytics Listener y Social Analytics Process.

El único fichero que hay que modificar para configurar la aplicación es:
src/main/webapp/WEB-INF/config.properties

Aquí añadiremos los datos de conexión con la base de datos (referencia de la base de datos se encuentra registrado en RUA). También añadiremos los datos de conexión con Elasticsearch. Cuando conozcamos la URL en la que se instalará esta aplicación, también la deberemos añadirla a este fichero. Además, en este fichero deberemos indicar las credenciales de acceso a las redes sociales Twitter e Instagram, que deben coincidir con las credenciales indicadas en *Social Analytics Listener v2.0*. Esto es necesario en caso de que se desee gestionar los datos descargados de los medios sociales mediante esta aplicación web. En el caso de no contar con estas credenciales, estos son los pasos a seguir para conseguirlas:

- Credenciales de Twitter para obtener tweets. Para ello hay que crear una cuenta de desarrollador en Twitter, siguiendo los pasos indicados en la página <https://dev.twitter.com/resources/signup>. Después hay que crear una nueva aplicación en la página <https://apps.twitter.com/app/new>. Una vez creada la aplicación copiaremos los tokens de acceso de la pestaña “Keys and Access Tokens” en el fichero “config.properties” de nuestra aplicación.
- Credenciales de Instagram para obtener posts. Aquí también es necesario crear una cuenta de desarrollador, siguiendo los pasos indicados en la página <https://www.instagram.com/developer/>. Después hay que crear un cliente nuevo en <https://www.instagram.com/developer/clients/register/>. Una vez creado el cliente, copiaremos todos los tokens de acceso del cliente que acabamos de crear y los copiaremos también en el fichero “config.properties”.

Por último, empaquetaremos la aplicación en un fichero WAR y la instalaremos en el servidor Tomcat, y ya estaría para su uso.